

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A method for providing access to a resource in a programming environment supporting concurrent tasks, comprising ~~the steps of:~~
 - (a) providing a latch to a first task, wherein the first task requests the latch to obtain access to the resource;
 - (b) accessing the resource with the first task~~releasing the latch after the first task has accessed the resource, while still holding the latch;~~
 - (c) marking the latch stealable rather than releasing the latch,~~while still holding the latch;~~ and
 - (d) marking the latch unstealable if the first task again requests the latch to obtain access to the resource prior to the latch being requested~~is not marked stolen by a second task seeking access to the resource.~~
2. (currently amended) The method of claim 1, ~~wherein the providing step~~ (a) further comprising ~~the steps of:~~
 - (a1) providing to the first task a latch marked stealable and held by a holding task; and
 - (a2) placing the resource in a consistent state prior to the first task accessing the resource.

3. (currently amended) The method of claim 12, further comprising ~~the step of~~ marking a set of flags for a latch stealable, wherein the resource is not placed in a consistent state because the latch is marked stealable, stolen, wherein the resource is placed in a consistent state and then accessed by the second task, or unstealable, wherein the first task again requests the latch to obtain access to the resource and the latch has not been requested by a second task seeking access to the resource, as requested by the latch.

4. (cancelled)

5. (currently amended) A computer readable medium containing program instructions for providing access to a resource in a programming environment supporting concurrent tasks, the program instructions for:

(a) providing a latch to a first task, wherein the first task requests the latch to obtain access to the resource;

(b) accessing the resource with the first task~~releasing the latch after the first task has accessed the resource, while still holding the latch;~~

(c) marking the latch stealable rather than releasing the latch,~~while still holding the latch;~~ and

(d) marking the latch unstealable if the first task again requests the latch to obtain access to the resource prior to the latch being requested ~~is not marked stolen~~ by a second task seeking access to the resource.

6. (original) The computer readable medium of claim 5, wherein the instructions for providing further comprising instructions for:

(a1) providing to the first task a latch marked stealable and held by a holding task; and

(a2) placing the resource in a consistent state prior to the first task accessing the resource.

7. (currently amended) The computer readable medium of claim 6, further instructions for marking a set of flags for a latch stealable, wherein the resource is not placed in a consistent state because the latch is marked stealable, stolen, wherein the resource is placed in a consistent state and then accessed by the second task, or unstealable, wherein the first task again requests the latch to obtain access to the resource and the latch has not been requested by a second task seeking access to the resource, as requested by the latch.

8. (cancelled)

9. (currently amended) A latch mechanism for a programming environment supporting concurrent tasks, comprising:

means for providing a latch for a resource to a first task, whereby the first task holds the latch;

~~means for the first task to release the latch while holding the latch;~~

means for the first task to mark the latch stealable rather than releasing the latch; and

means for the first task to ~~mark~~ remark the latch unstealable if the first task again requests the latch to obtain access to the resource prior to the latch being requested ~~where the latch is not marked stolen~~ by a second task seeking access to the latch.

10. (original) The latch mechanism of claim 9, wherein the means for providing a latch to the first task further comprising:

means for the first task to be provided, on request, with a latch marked stealable and held by a holding task, and means for placing any resources associated with the latch in a consistent state prior to the first task accessing the resources.

11. (currently amended) The latch mechanism of claim ~~9~~ 10, further comprising a set of flags for marking a latch stealable, wherein the resource is not placed in a consistent state because the latch is marked stealable, stolen, wherein the resource is placed in a consistent state and then accessed by the second task, or unstealable, wherein the first task is again provided with the latch and the latch has not been provided to a second task seeking access to the resource, by a task.

12. (cancelled)

13. (cancelled)

14. (cancelled)

15. (cancelled)